METHOD AND APPARATUS FOR ASSESSING THE STATUS OF WORK WAITING FOR SERVICE

FIELD OF THE INVENTION

The present invention is directed to the scheduling of work items in a resource allocation system. In particular, the present invention is directed to service time goals for work items in a queue awaiting service, and assessing the status of work items in queue relative to their service time goals.

BACKGROUND OF THE INVENTION

In present day automatic contact distribution (ACD) systems, resource selection and allocation algorithms are commonly employed to perform calculations related to timing of operations and service time goals for work items in queue. These calculations are performed, for example, when a telephone call is received at a call center. When such a call is received, it is typically assigned to a pool of resources responsible for answering telephone calls. Furthermore, such calls generally have a service time goal, such as three minutes, which is the goal for an agent to answer the call. These service time goals are useful to help ensure a customer is not waiting to speak to an agent for a long period of time, which may reduce customer satisfaction.

In such ACD systems, contacts incoming to a contact center are answered and handled by a plurality of resources. The ACD system automatically distributes and connects incoming contacts to whatever resource, generally agents, have the skill set suited to handle the contacts and are free, *i.e.,* not handling other contacts at the moment. As used herein, a

"contact" refers to any mode or type of contact between two entities, including without limitation voice calls, VoIP, text-chat, e-mail, fax, electronic documents, web forms, voice messages, and video calls, to name but a few.

The contacts are placed in different queues based upon preestablished criteria, such as business/service policies, objectives, and goals for each contact type, and are typically placed in each queue in the order of their arrival and/or priority. Due to the random and peaked nature of inbound contacts from customers, a contact center may become overloaded when no suitable resources are available to handle contacts as the service time goal for the contacts expires. Furthermore, a contact center may have sufficient resources to handle present contacts which have service time goals currently expiring, but may not have enough resources to handle the contacts which have a service time goal at some point in the future.

As is known in the art, it is common for such ACD systems to include algorithms which detect whether service time goals are being met, and also to predict if service time goals are likely to be met in the future. Numerous techniques have been devised for determining an actual or anticipated wait time for each queued item, and the queued items are typically serviced based on the actual and/or anticipated wait time. However, such techniques generally look at only the head of the queue, in order to determine if the contact center is currently behind target, or is in a state of immediate risk. Techniques which are used to predict of service time goals are likely to be met in the future generally only look at the tail of the queue, or the last item in the queue, and make a determination of whether it is predicted that this item will be serviced at or before the service time goal for that work item,

and give a yes/no answer as to whether there is a future risk. As will be understood, the last item in the queue may follow a number of items which all have a service time goal which will expire at substantially the same time. Thus, it is possible that the last item in a queue will show no future risk, while there in actuality is a future risk associated with the relatively heavy workload which precedes the last work item in the queue. Accordingly, it would be advantageous to have a method and apparatus which is able to determine future risk, and also determine when such risk will arise and the amount of resources required to correct the potential shortfall in resources.

Another problem with such techniques is that they were designed for real time servicing. As mentioned above, it is common to receive contacts in the form of e-mail, fax, electronic documents, web forms, voice messages, which do not require immediate attention of an agent, but rather are required to be attended to within a preset service time goal period. For example, the contact center may have a service time goal for electronic mail of one business day. Likewise, web forms which are received may have a goal of being answered within two business days. Such contacts are referred to as "back office contacts," which are placed into an queue which is to be serviced by a back office, meaning that they are not serviced by agents in real time with a contact.

In some ACD systems, such work items are placed in unordered work queues. These items often are received at different times, and have different service goals. Thus, if the items were to be placed in an ordered work pool, with the contacts ordered by the amount of time left to service the contact, each time a contact is added to the queue, the queue may

have to be reordered. As will be understood, reordering a work queue can consume a significant amount of resources in a system, thus it may be advantageous to place such work items in an unordered work pool or work queue. One problem with placing contacts in an unordered work pool is that resource allocation algorithms are generally designed to operate using ordered work pools. Thus, it would be advantageous to have an unordered work pool with resource allocation algorithms which are able to assess the work in the unordered work pool to determine a status of the pool and make predictions regarding potential future resource shortfalls for the work pool.

SUMMARY OF THE INVENTION

These and other needs are addressed by the various embodiments and configurations of the present invention. The present invention is directed generally to a method and apparatus for assessing the status of work awaiting service in an ordered or unordered work pool. The methodology is particularly useful in contact centers.

In one embodiment, work items, such as contacts, product orders, service requests, are placed into a work queue for service by a resource. The status of work waiting for service is assessed by generating, based at least in part on the work queue, an ordered set of items related to the work items in the work queue, and analyzing the ordered set to predict a future state of the work queue. A required queue position (RQP) for each work item in said work queue, may be determined, the RQP based on a service time goal for each work item and an estimated time for completion of work items. The ordered set of items may be

generated by creating an array of counters, each element in the array of counters corresponding to a predefined range of required queue positions. The array of counters may be modified by incrementing a counter in the array of counters associated with the RQP for each work item. The required queue position is determined, in one embodiment, for each work item, by subtracting an amount of time since the work item was received from the service time goal for the work item to obtain a remaining time for the work item. For each work item, determining a required queue position may include dividing the remaining time for the work item by the weighted advance time (WAT) of the work queue. The weighted advance time is the measure of the average time that is required for a work item to advance one position in the queue. The calculation of weighted advance time is described in U.S. Patent No. 5,505,898, the disclosure of which is incorporated herein by reference in its entirety.

The ordered set of items, in one embodiment, is generated by determining a range of required queue positions which correspond to each item within the ordered set, and incrementing a counter associated with the item within the ordered set which corresponds to a required queue position associated with each work item. The range of queue positions for each item in the ordered set may be set to preestablished criteria, such as, for example, each item in the ordered set may correspond to one queue position, or each item in the ordered set, where the number of the item is N, may be $2^{N-1} < \text{RQP} \geq 2^N$.

Analyzing the ordered set to predict a future state of the work queue, in one embodiment, includes the steps of creating an index variable, setting the index variable to

one, creating a sum variable, setting the sum variable to zero, calculating a new sum as the sum of the previous value of the sum variable and the value of the item in the ordered set which corresponds to the index variable, determining if the sum is greater than the index, setting a state to "Future Risk" when the sum is greater than the index, and incrementing the index and repeating the calculating a new sum, determining if the sum is greater than the index, and setting a state when the sum is not greater than the index. In another embodiment, a range of queue positions corresponds to each item within the ordered set, and the determining step includes determining if the sum is greater than the highest number queue position which is associated with the item in the ordered set which corresponds to the index variable.

The analyzing of the ordered set may further include determining if there are additional items in the ordered set, and setting a state to "On Target" when there are no additional items in the ordered set. The analyzing of the ordered set may also include, when the sum is greater than the index, predicting a time of the "Future Risk". The time may be calculated as the product of the index and the estimated time for completion of work items. The analyzing of the ordered set may include, when the sum is greater than the index, determining an extent of the "Future Risk". The extent of the future risk is calculated, in one embodiment, as the difference between the sum and the index. The extent of the future risk is calculated, in another embodiment, as the difference between the sum and the highest number queue position associated with the item in the ordered set which corresponds to the index variable. In another embodiment, the invention provides a computer readable medium

containing instructions for performing the steps for determining the status of work waiting for service, and a logic circuit operable to perform the steps for determining the status of work waiting for service.

The invention also provides, in another embodiment, a computational component for performing a method, the method comprising: determining a required queue position (RQP) for each of a plurality of work items, the RQP based on a remaining time for the work item and a weighted advance time for the work queue incrementing a counter in an element of an array of counters, the element corresponding to a predefined range of required queue positions; and analyzing the array of counters to predict a future state of the work items. The determining a required queue position step may include, for each work item, subtracting an amount of time since the work item was received from a service time goal for the work item to obtain a remaining time for the work item. The determining a required queue position step may include determining the weighted advance time for the work queue and for each work item, dividing the remaining time by the weighted advance time for the work queue. The incrementing a counter step may include determining a range of required queue positions which correspond to each element within the array of counters, and incrementing a counter associated with the element within the array of counters which corresponds to the required queue position obtained in the determining a required queue position step. In one embodiment, the predefined range of queue positions for each element in the array of counters is one. In another embodiment, the predefined range of queue positions for each element in the array of counters, where the number of the element is N, is $2^{N-1} < \text{RQP} \geq 2^N$.

In one embodiment, the analyzing step includes the steps of: creating an index variable; setting the index variable to one; creating a sum variable; setting the sum variable to zero; calculating a new sum as the sum of the value of the sum variable and the value of the counter in the element of the array of counters which corresponds to the index variable; determining if the sum is greater than the index; setting a state to "Future Risk" when the sum is greater than the index; and incrementing the index and repeating the calculating a new sum, determining if the sum is greater than the index, and setting a state when the sum is not greater than the index. In one embodiment, the determining step includes determining if the sum is greater than the highest number queue position which is associated with the element of the array of counters which corresponds to the index variable. The analyzing step may also include: determining if there are additional elements in the array of counters; and setting a state to "On Target" when there are no additional elements in the array of counters. The analyzing step may also include, when the sum is greater than the index, determining a time of the "Future Risk". The time of the future risk is calculated as the product of the index and the weighted advance time of the work queue. The analyzing step may further include, when the sum is greater than the index, determining an extent of the "Future Risk". The extent may be calculated as the difference between the sum and the index. In another embodiment, the analyzing step further includes, when the sum is greater than the highest number queue position associated with the element of the array of counters which corresponds to the index variable, determining an extent of the "Future Risk," calculated as

the difference between the sum and highest number queue position associated with the element.

Another aspect of the invention provides a table maintained in an electronic memory of a contact center, comprising an identity of at least two work items, and an ordered list having entries associated with a predefined range of required queue positions for the work items. The entries, in one embodiment, indicate required queue positions for the work items. The predefined range of required queue positions for each entry in the ordered list, in one embodiment is one. In another embodiment, the predefined range of required queue positions for each entry in the ordered list, where the number of the entry is N, is $2^{N-1} < RQP \geq 2^N$.

Another aspect of the invention provides a contact center for servicing a plurality of contacts received from a plurality of customers, comprising: a plurality of workstations corresponding to a plurality of resources; a central server in communication with the plurality of workstations, comprising at least one queue of contacts, each of the contacts having an associated service time goal, and a workload monitoring agent operable to (a) monitor the queue of contacts; (b) assess a state of the queue of contacts with respect to the service time goals for the plurality of contacts; and (c) determine a number of contacts which are likely to not meet their service time goals and a time at which the service time goal for the number of contacts will expire. In one embodiment, the contacts in the queue comprise one or more of real time and non-real time contacts. In another embodiment, the workload monitoring

agent is further operable to identify a weighted advance time of the work queue and determine a required queue position for each of the contacts The workload monitoring agent may determine the required queue position based on the weighted advance time of the work queue, an elapsed time since the contact was received at the queue, and a service time goal for the contact. The required queue position is calculated, in an embodiment, as the difference between the service time goal and the elapsed time divided by the weighted advance time of the work queue. The contacts within the plurality of contacts may have at least two service time goals. The workload monitoring agent is further operable to determine, in an embodiment, a representation of required queue positions associated with the contacts in the queue. In one embodiment, a predetermined workload level exists when a queue position in the representation of required queue positions is less than a number of enqueued contacts ahead of the queue position in the representation of required queue positions. The time which the predetermined workload level will likely exist is the product of the weighted advance time of the work queue and queue position at which the predetermined workload level will likely exist. The number of contacts required to be serviced is the difference between the required queue position and the number of enqueued contacts before the required queue position.

These and other advantages will be apparent from the disclosure of the invention contained herein, particularly when taken in conjunction with the attached drawings.

The above-described embodiments and configurations are neither complete nor exhaustive. As will be appreciated, other embodiments of the invention are possible

utilizing, alone or in combination, one or more of the features set forth above or described in detail below.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a contact center of one embodiment of the present invention;

Fig. 2 is a plot of work item volume (vertical axis) versus time (horizontal axis);

Fig. 3 is flow chart diagram illustrating the operational steps for calculating required queue positions for work items of one embodiment of the present invention;

Fig. 4 is a flow chart diagram illustrating the operational steps for analyzing an array of counters of one embodiment of the present invention; and

Fig. 5 is a table illustrating elements of an array of counters and ranges of required queue positions associated with each element according to one embodiment of the present invention.

DETAILED DESCRIPTION

Fig. 1 shows an illustrative embodiment of the present invention.

A contact center 6 comprises a central server 10 (such as a Definity™ or Multi-Vantage™ Enterprise Communications Server running modified Advocate™ software of Avaya, Inc.), a set of data stores or databases 12 containing contact or customer related information and other information that can enhance the value and efficiency of the contact, a plurality of servers, namely a fax server 24, a data network server 20, an email server 16, and other servers 13, a private branch exchange PBX 28 (or private automatic exchange PAX), a first plurality or set of resources 14 (which are shown as being human agents) operating computer work stations 15, such as personal computers, and/or telephones 17 or other type of voice communications equipment, all interconnected by a local area network LAN (or wide area network WAN) 36, and a second plurality or set of resources 100 (which are shown as being human agents) also operating computer work stations 15, such as personal computers and/or telephones 17 or other types of voice communications equipment, connected to the PBX 28 via a public switched telephone network or PSTN 48 and to the central server 10 via a data network 44, such as the Internet. The fax server 24, web server 20 and email server 16 are connected via communication connections 40 to the data network 44.

The other servers 13 can be connected via optional (dashed) communication lines 22, 32 to the PBX 28 and/or the data network 44. As will appreciated, other servers 13 could include a scanner (which is normally not connected to the PBX 28 or network 44), interactive

voice recognition IVR software, VoIP software, video call software, voice messaging software, an IP voice server, and the like. The PBX 28 is connected via a plurality of trunks 18 to the PSTN 48 and to the fax server 24 and telephones 17 of the resources 14. As will be appreciated, faxes can be received via the PSTN 48 or via the network 44 by means of a suitably equipped personal computer. The PBX 28, fax server 24, email server 16, web server 20, and database 12 are conventional.

As will be appreciated, the central server 10 is notified via LAN 36 of an incoming realtime or non-realtime contact by the telecommunications component (*e.g.*, PBX 28, fax server 24, email server 16, web server 20, and/or other server 13) receiving the incoming contact. The incoming contact is held by the receiving telecommunications component until the central server 10 forwards instructions to the component to forward the contact to a specific workstation and/or resource. The server 10 distributes and connects these contacts to workstations of available resources based on a set of predetermined criteria. The resources process the contacts sent to them by command of the central server 10.

In the architecture of Fig. 1 when the central server 10 forwards a realtime contact such as a telephone call to a resource, the central server 10 also forwards information from databases 12 to the resource's computer work station for viewing (such as by a pop-up display) to permit the resource to better serve the customer. The information is typically effected by the establishment of a data communications link between the central server and the target resource's workstation.

In one configuration, the first and second pluralities or sets of resources correspond, respectively, to employees and nonemployees of the business or enterprise operating the contact center. For example, the second plurality or set of resources can be contractors, subcontractors, employees of another organization (including a bidding house), and the like. The first plurality of resources are served directly or supported by the central server/PBX and commonly service contacts to the center. In other words, the first plurality of resources or set of resources/workstations are subscribers to the enterprise network defined by the contact center 6 or are within the premises serviced by the server/PBX. The second plurality or set of resources/workstations are generally not served and/or supported directly by the central server and are typically geographically dislocated from the first plurality or set of resources. In other words, the second plurality of resources or set of workstations/resources are not subscribers to or supported by the enterprise network and are external to the premises serviced by the PBX and central server. The second set of resources may thus be "external" in that they are not directly supported as terminal endpoints by the server PBX (e.g., they do not have an extension associated with an internal endpoint serviced by the switch/server). Communications with these resources are directed through the PSTN 48 (for telephone calls) (and are received at an external port of the switch/server) and through the data network 44 (for data communications such as customer-related information transmission). The second set of resources may be used to augment or support the first set of resources, such as by servicing less valuable or profitable work items through, for example, a bidding type process discussed in copending U.S. Patent Application "Contact Center Resource Allocation Based

On Work Bidding/Auction", filed on even date herewith, to Flockhart et al., which is incorporated herein by this reference.

The central server 10 includes a memory 30 having a plurality of first sets 38 of contact queues 42 and 46 corresponding to the first plurality of resources. Each set of contact queues conventionally serves and holds contacts (or work items) for a different work type and/or for realtime versus non-realtime contacts. In the depicted embodiment, queues 42 serve non-real-time contacts while queues 46 serve real-time contacts. This embodiment is particularly suited for a Customer Relationship Management (CRM) environment in which customers are permitted to use any media to contact a business. In a CRM environment, both realtime and non-realtime contacts must be handled and distributed with equal efficiency and effectiveness. Within each set of queues, each queue holds contacts of a different priority and/or different type (*e.g.*, e-mail, fax, electronic or paper documents, webform submissions, voice messages, voice calls, VoIP calls, text chat, video calls, and the like). The priority of a contact is determined according to well known predefined criteria. Each queue may function as a first-in, first-out (FIFO) buffer memory, and include a plurality of items, or positions 50, each for identifying a corresponding one enqueued contact. The position at the head of the queue is considered to be position 1, the next subsequent position to be position number 2, and so forth. The queues may also be unordered work queues, in which the work items contained in the queues are not in a FIFO memory. Such an unordered configuration may be beneficial, for example, for a non-real-time queue having work items with differing

service time goals where re-ordering the work queue following the completion of each work item would take a significant amount of resources.

Memory 30 further includes a wait time determining agent 54. As its name implies, this agent determines an estimate of how long a contact that is placed in a queue will have to wait before being delivered to a resource for servicing and/or has already waited for servicing. The estimate may be derived separately by the agent 54 for each work item in each queue of each set, and is referred to herein as an estimated wait time (EWT).

For realtime contacts, the EWT is based on any suitable algorithm, such as the average rate of advance of contacts through positions 50 of the contacts' corresponding queue referred to herein as a weighted advance time (WAT). This estimate is derived separately by the agent 54 for each queue. An illustrative implementation of the agent 54 for real-time contacts is disclosed by U.S. Patent 5,506,898, which is incorporated herein by this reference.

For non-realtime contacts, the EWT estimate is generally determined differently than for realtime contacts. One approach for calculating the wait time is set forth in U.S. Patent Application Serial No. 09/641,403, filed August 17, 2000, entitled "Wait Time Prediction Arrangement for Non-Real-Time Customer Contacts", which is incorporated herein by this reference.

Memory 30 can further include a work item selecting agent 26. Agent 26 is conventional. It selects a work item from one or more of the queues to be serviced by an

available resource based on wait time and/or business/service policies, objectives, and goals for each contact type.

The memory further includes a workload monitoring agent 70, as will be discussed below, for predicting potential deficiencies or surpluses for the first set of resources. A bid item selecting resource 74, as discussed in copending U.S. Patent Application "Contact Center Resource Allocation Based On Work Biddine/Auction", filed on even date herewith, to Flockhart et al., may be used if a deficiency in the first set of resources is predicted, for configuring and tracking a bidding process for each work item and selecting the winning bidder for such work items.

The workload monitoring agent 70 receives EWT information from the wait time determining agent 54, monitors the length of each queue, the numbers of available resources in the plurality of first resources, the types and priorities of contacts in each monitored queue, and anticipated workload levels. Based on this information, the workload monitoring agent 70 predicts when the contact center must take action in order to meet predetermined business/service policies, objectives, and goals for each contact type.

A graphical illustration of prediction of the amount of work items completed as a function of time is contained in Fig. 2. As illustrated in Fig. 2, the rate at which work items can be handled as a function of time is represented by line 104. The sinusoidal waveform 108 represents the number of work items that must be serviced by the resources as a function of time. In the example illustrated in Fig. 2, there is a surplus of available resources earlier than time t_2, and a surplus of work items after time t_2. If the workload monitoring agent 70

predicts a surplus work item condition at time t_1, actions may be taken that are associated with excess resources, such as making number of surplus resources available for other uses. If the workload monitoring agent 70 predicts a surplus of work items, actions associated with a deficiency in resources may be taken such as a bidding process in which the surplus work items are assigned to one or more of the members of the second set of resources no later than time t_2.

With reference now to Figs. 3-4, the operation of the workload monitoring agent 70 of one embodiment of the present invention is described in more detail. As discussed above, work queues 42, 46 contain work items which are to be serviced by a particular pool of resources. As mentioned, in the embodiment of Fig. 1, work queues 42 are associated with non-real-time contacts and may include more than one type of contact, with each contact having a differing service time goal. Non-real time contacts may be "back office" type contacts which include contacts of several different types. For example, items in work queues 42 may include email enquiries, web forms, and purchase requests, among other things. Email enquiries may have a service time goal of one hour, web forms may have a service time goal of four hours, and purchase requests may have a service time goal of one day. These contacts, in one embodiment, are placed in work queues 42 in the order in which they were received. Accordingly, the first item in a queue may not be the item which has the shortest amount of time before the expiration of the service time goal. Such a situation may arise when the first item on a queue is a purchase request (having a service goal of one day), and the second item on the queue is an email enquiry (having a service goal of one hour).

In one embodiment, the workload monitoring agent 70 assesses each of the work queues 42, 46, and assigns a state to each of the work queues 42, 46 to indicate if the queue is "behind target," "on target," has an "immediate risk," or has a "future risk." Furthermore, if the queue is in the behind target, immediate risk, or future risk states, the workload monitoring agent 70 is operable to make a prediction as to the amount of resources required to solve the problem. If a future risk state is predicted, the workload monitoring agent 70 is also operable to make a prediction of when the system will enter the risk state.

When determining the state of the work queues 42, 46, the workload monitoring agent 70, on an event basis and/or on a periodic basis, will examine a work queue or pool to determine the current state of the work items relative to their individual service time goals. Each work item in the queue or pool will have a service time goal, as described above. The remaining time for each work item in the queue or pool may be calculated as:

Remaining time = (Service Time Goal) – (Current Time).

If any work item has a negative remaining time, then the state of the work item and the work queue is "behind target." In such a situation, in one embodiment, the workload monitoring agent 70 generates an appropriate message to indicate the behind target state. The message may be sent to appropriate personnel or software system that may then take actions to remedy the problem. The message may include electronic notifications to appropriate personnel, display on a user interface, or other indication. The workload monitoring agent 70, in one embodiment, then proceeds to calculate required queue positions for each of the work items in the work queue. In another embodiment, if a behind target state is detected, the workload

monitoring agent 70 simply generates the appropriate alert and does not proceed to calculate required queue positions.

When determining the required queue position, WAT for the queue is determined. As mentioned above, the WAT is the average elapsed time between each service event from the queue. For example, if work items are serviced from the queue at an average rate of one every six seconds, the WAT for that queue is six seconds, meaning that, for example, the second item in the queue would be predicted to be completed in 12 seconds. Once the WAT is determined, the required queue position for each work item in the queue or pool is then calculated as the remaining time for the work item divided by the WAT, and rounded down to the next integer number. For example, if a work item has 20 seconds remaining before its service time goal expires, and the WAT for the queue or pool is six seconds, the prediction is that the work item must be serviced by one of the next three agents to become available to service work from this queue in order to meet the service time goal. Thus, the required queue position for the work item is three.

Referring now to Fig. 3, a flow chart diagram illustrating the operational steps for calculating required queue positions for each of the work items in the queue is now described. An ordered list associated with the work items in the work queue or pool is created, which in one embodiment is an array of counters referred to as the Required Queue Position Array ("RQPA"). Each element in the RQPA indicates the number of work items which are required to be completed during the required queue position. The required queue position is indicated by the position of the element in the RQPA. Thus, the second element

in the RQPA indicates the number of work items with a required queue position of two. A counter, independent of the RQPA, referred to as the "Behind Target Count," is also associated with the work queue or pool, and indicates the number of work items which are behind target. The assessment of the work queue is initiated, as indicated at block 150. Each element of the RQPA is initialized to zero, and the Behind Target Count is initialized to zero, as indicated by block 154. This is accomplished by setting each element in the RQPA, and the Behind Target Count, to have a value of zero. At block 158, the first item in the work queue is scanned. The required queue position ("RQP") is calculated for the work item, as noted by block 162. Following the calculation of the required queue position for the work item, it is determined whether the required queue position is less than zero, as noted by block 163. A negative required queue position indicates that the remaining time is negative, and the work item is behind target. If the required queue position is negative, the Behind Target Count is incremented, according to block 164. If the required queue position is zero or greater, the element in the RQPA array corresponding to the required queue position, RQPA[RQP], is incremented, according to block 166. At block 170, it is then determined if there are more work items in the work queue. If there are no more work items, it is determined if the Behind Target Count is greater than zero at block 171. If the Behind Target Count is not greater than zero, the RQPA is analyzed, as noted by block 174. If the Behind Target Count is greater than zero, the workload monitoring agent sets the current state to be "behind target," indicating that the time of the problem is the current time, and that the extent of the problem is the Behind Target Count, according to block 172. If there are more work

items in the work queue, the next work item in the work queue is scanned, according to block 178. Following the scan of the next work item, the operational steps described with respect to blocks 162 through 178 are repeated. Thus, following the assessment of the work queue, the RQPA is created which includes items in each array element corresponding to the number of work items that must be completed by the time corresponding to the queue position of the RQPA.

Referring now to Fig. 4, the operational steps for analyzing the RQPA are described for one embodiment of the present invention. Initially, as noted by block 200, it is determined if element zero of the required queue position array (RPQA[0]) is greater than zero. This indicates that one or more work items has a required queue position of zero, which indicates that the work item should have been serviced by the previous agent to become available. By the time the next agent becomes available that can process the work item, it is likely that the work item will already have missed its service time objective. In such a case, the workload monitoring agent sets the current state to be "immediate risk," indicating that the time of the problem is the current time, and that the extent of the problem is the value of RQPA[0], as indicated by block 204. If it is determined that RQPA[0] is zero, the workload monitoring agent 70, at block 208, sets an index variable to one, and a sum variable to zero. The workload monitoring agent 70 then sets the sum variable to be the total of the sum, plus the value of the required queue position array element corresponding to the index variable (Sum = Sum + RQPA[Index]), as noted by block 212. At block 216, it is then determined if the sum is greater than the index. If the sum is greater than the index, this

indicates that the total number of work items which need to be completed is greater than the total number of work items that are predicted to be completed at the current rate of service work items as measured by the WAT.

If the sum is greater than the index, the state of the system is set to "Future Risk," the predicted time of the problem is the product of the index and the WAT (Index*WAT), and the extent of the problem is the difference between the sum and index (sum-Index). If the sum is not greater than the index at block 216, the index is incremented, according to block 224. The determination is made at block 228 if the end of the RQPA has been reached. If the end of the RQPA has been reached, the state of the system is "On Target," meaning that there are no predicted shortfalls in resources for the work items that are in the work queue or work pool. If at block 228, it is determined that the end of the RQPA has not been reached, the operational steps of blocks 212 through 228 are repeated until either the end of the RQPA is reached, or until a state of "Future Risk" is found. For example, if the index is at six, and the sum of the work items with required queue positions of six or less is nine (Index=6, Sum=9), this indicates that three work items are predicted to not be completed by their service time goal. If the WAT is 15 seconds, the time that this problem will occur is predicted to be 90 seconds (Index * WAT = 6 * 15 = 90 seconds). The extent of the problem is predicted to be three work items. This information, in one embodiment is displayed on a user interface, and an electronic notification is forwarded to appropriate personnel or software systems that may then attempt to correct the problem before it occurs. While the embodiment of Fig. 4 stops when evidence of a problem is detected, in another embodiment

the entire array is scanned, even if a "Future Risk" state is found, in order to find all potential problems and/or the most severe problem.

Furthermore, in one embodiment, the workload monitoring agent 70 monitors the work queues for instances in which surplus resources are predicted. For example, if the state of the system is "on target," the workload monitoring agent 70 may determine how many surplus resources are available. Such surplus resources may be allocated to different functions, for example. When determining the number of surplus resources, and the time of the surplus, similar calculations as described above may be utilized. The difference between the sum and the highest number queue position associated with an element in the RQPA may be utilized to indicate how many surplus resources are predicted for the time associated with the queue position. For example, if the sum is 20 and the highest queue position associated with the RQPA element is 30, this indicates that ten surplus resources are predicted to be available at that point in time while maintaining service time goals for work items in the work queue. If the state of the system is "on target," this indicates that these surplus resources may be used for other tasks while maintaining service time goals for all of the work items presently in the queue if removing the surplus resources would not place the state of the system to future risk.

As can be seen, the system and method provided above produces results of both a time of the potential problem, as well as the extent of the potential problem, which is beneficial in proactive solving of the potential problem. Furthermore, this system and method may be used in both ordered work queues, as well as unordered work queues or work

pools. This is a result of each work item being evaluated for its required queue position, and the array of counters having elements which correspond to the required queue positions which are incremented each time a work item having that required queue position is scanned. Accordingly, even if work items are unordered, when each item is scanned, its required queue position is included in the array of counters which may then be assessed to determine the status of work queue or work pool.

It will be understood that the embodiment of Figs. 3 - 4 is illustrative of one of many techniques which may be used to assess the status of work items waiting for service. Other alternatives exist for creating and analyzing an ordered list associated with work items waiting for service, and would be readily understood by one of skill in the art. For example, a required queue position array may be initialized with each element within the array having a numerical value equivalent to the position of the element within the array. Thus, position one of the array would have a value of one, position two of the array would have a value of two, and so on. When a work item is evaluated and the required queue position for the work item is determined, the value of the elements in the required queue position array associated with the required queue position and higher may then be decremented. In this manner, the elements within the required queue position array are updated to indicate a status of the work waiting for service. The RQPA may then be scanned for any negative items, which would indicate that more work items are present than are anticipated to be completed prior to the time associated with that required queue position.

In another embodiment, the array of counters does not include every required queue position as an element of the array, but rather a range of required queue positions are assigned to particular array elements. Fig. 5 is a table illustrating an array of counters 250, and corresponding queue positions 254 associated with each element 258 of the array. In this embodiment, for the elements in the array of counters, each element 258 ("N") in the array stores the number of work items having a required queue position ("RQP") in the range of: $2^{N-1} < RQP \geq 2^{N}$. Thus, as illustrated in the table of Fig. 5, for example, array element three would contain a count of the number or work items having a RQP in the range of five through eight, and array element four would contain the count of the number of work items having a RQP in the range of 9 through 16. When scanning work items in this embodiment, any work items which have an RQP of zero will place the system into an "immediate risk" state, and the array is then analyzed to determine any future risk states.

The analysis of the array can detect potential problems in the defined range of RQPs, rather than at every RQP. If work queues are of a significant size, this embodiment results in a performance increase due to the array being much shorter, thus consuming fewer system resources. In this embodiment, an array of counters containing ten elements can handle work queues or work pools of about 1000 work items. Additionally, this embodiment places greater focus on work items having service time goals which expire in a relatively short amount of time, where there is less time to react to any detected problems, while problems with work items having service time goals which expire in a relatively long amount of time are indicated in an approximate time range. The increased granularity of lower required

queue positions allows for immediate attention to short term problems, while the decreased granularity of higher required queue positions allows for longer term planning and solutions. Thus, if a potential problem is detected in, for example, required queue positions 65-128, appropriate personnel may be alerted who then have a relatively long period of time to take corrective action compared to a situation where a potential problem is detected in required queue position 4. While the embodiment of Fig. 5 uses RQP ranges which are powers of two, any ranges of RQPs may be used for elements in such an array, as will be readily understood by one of skill in the art.

The foregoing discussion of the invention has been presented for purposes of illustration and description. Further, the description is not intended to limit the invention to the form disclosed herein. Consequently, variations and modifications commensurate with the above teachings, within the skill and knowledge of the relevant art, are within the scope of the present invention. The embodiments described hereinabove are further intended to explain the best mode presently known of practicing the invention and to enable others skilled in the art to utilize the invention in such or in other embodiments with various modifications required by their particular application or use of the invention. It is intended that the appended claims be construed to include the alternative embodiments to the extent permitted by the prior art.